Informatik 2 für Regenerative Energien

Klausur vom 4. April 2022[c1]: Lösungen

[c1] jl: 23. September 2019

Jörn Loviscach

Versionsstand: 15. August 2022, 12:46

1. Die Fehler:

Zeile	korrekter Programmtext
27	`DateTime zeitpunkt;`
51	`public Geokoordinaten(double breitengrad, ...`
58	`class PositionAmFluss`
79	`if (fluss != p.fluss)`
85	`return Math.Abs(flusskilometer - p.fluss...`
102	`class Station`
112	`protected List<Messung> messungen = new ...`
116	`{ get { return warnungen.Count;  } }`
127	`void Warne(string grund)`
137	`foreach (Station station in stationen)`
139	`if (station.positionAmFluss.LiegtFlussabwärtsVon(`
155	`: base(geokoordinaten, positionAmFluss)`
162	`messungen.Add(new Pegelstandsmessung(...`
164	`if (pegel > kritischerPegel)`
166	`WarneAlleStationenFlussabwärts();`

2. `x` ist `1`, weil der Kölner Pegel über dem kritischen Wert gelegen hat und flussabwärts Duisburg gewarnt worden ist. `y` ist `0`, weil flussaufwärts an der Lutter keine Warnung gegeben worden ist. `z` ist `100.0`, weil dies der Betrag der Differenz der Flusskilometer ist.

3. *Zum Beispiel:* In der Klasse `Position` diese Property ergänzen:

```
public double Flusskilometer
    { get { return flusskilometer; } }
```

Und am Anfang des Konstruktors der Klasse `Station` dies ergänzen:

```
if (positionAmFluss.Flusskilometer > 3000.0)
```

```
{
throw new ApplicationException(
    "Unplausibel hohe Zahl an Flusskilometern.");
}
```

4. *Zum Beispiel:* In der Klasse `Messung` einen öffentlichen Getter für den Zeitpunkt und in der Klasse `Pegelstandsmessung` einen öffentlichen Getter für den Pegel ergänzen und dann die gefragte Methode so implementieren:

```
double BestimmeMaximalpegelSeit(DateTime wann)
{
    bool gefunden = false;
    double wert = double.MinValue;
    foreach (Messung m in messungen)
    {
        if (m.Zeitpunkt >= wann && m is Pegelstandsmessung)
        {
            gefunden = true;
            Pegelstandsmessung p = (Pegelstandsmessung)m;
            wert = Math.Max(wert, p.Pegel);
        }
    }

    if (gefunden)
    {
        return wert;
    }
    else
    {
        return double.NaN;
    }
}
```

Oder eleganter dies als Körper der obigen Funktion:

```
var ms = messungen.Where(m => m is Pegelstandsmessung
                              && m.Zeitpunkt >= wann);
if (ms.Count() == 0)
{
    return double.NaN;
}
else
{
    return ms.Min(m => ((Pegelstandsmessung)m).Pegel);
}
```

5. *Zum Beispiel:*

```
class PegelUndFließgeschwindigkeitsmessung
    : Pegelstandsmessung
{
    double fließgeschwindigkeit;

    public PegelUndFließgeschwindigkeitsmessung(
    DateTime zeitpunkt, double pegel,
double fließgeschwindigkeit)
    : base(zeitpunkt, pegel)
    {
        this.fließgeschwindigkeit = fließgeschwindigkeit;
    }
}
```

6. *Zum Beispiel:*

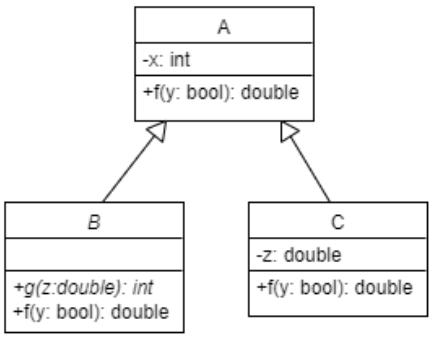
```
public static int BestimmeGesamtzahlWarnungen()
{
    int gesamtzahl = 0;
    foreach (Station station in stationen)
    {
        gesamtzahl += station.warnungen.Count;
    }
    return gesamtzahl;
}
```

   Oder eleganter dies als Körper der obigen Funktion:

```
return stationen.Sum(s => s.warnungen.Count);
```

   Diese Methode *kann* statisch gemacht werden, weil sie nur über die statische Variable `stationen` auf die Instanzen zugreift. Sie *sollte* statisch gemacht werden, weil sie keine bestimmte Instanz betrifft.

7. Das Diagramm:



8. Die Werte sind 3, 4, 1.